12

AMRL-TR-79-88

LEVEL

AD A088205

COMPUTER PROGRAM TO ADD NOISEMAP GRIDS OF DIFFERENT SPACINGS

HARRY SEIDMAN

BOLT BERANEK AND NEWMAN INC.
21120 VANOWEN STREET
CANOGA PARK, CALIFORNIA 91303

APRIL 1980

Approved for public release; distribution unlimited.

AIR FORCE AEROSPACE MEDICAL RESEARCH LABORATORY
AEROSPACE MEDICAL DIVISION
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

NOTICES

When US Government drawings, specifications, or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Please do not request copies of this report from Aerospace Medical Research Laboratory. Additional copies may be purchased from:

National Technical Information Service
5285 Port Royal Road
Springfield, Virginia 22161

Federal Government agencies and their contractors registered with Defense Documentation Center should direct requests for copies of this report to:

Defense Documentation Center
Cameron Station
Alexandria, Virginia 22314

TECHNICAL REVIEW AND APPROVAL

AMRL-TR-79-88

This report has been reviewed by the Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

Henning E. von Gierke

HENNING E. VON GIERKE
Director
Biodynamics and Bioengineering Division
Aerospace Medical Research Laboratory

AIR FORCE/56780/27 June 1980 — 150

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE	READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER: AMRL-TR-79-88 / 2. GOVT ACCESSION NO.: AD-A088 205	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): COMPUTER PROGRAM TO ADD NOISEMAP GRIDS OF DIFFERENT SPACINGS.	5. TYPE OF REPORT & PERIOD COVERED: Final Report
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Harry Seidman	8. CONTRACT OR GRANT NUMBER(s): F33615-79-C-0501
9. PERFORMING ORGANIZATION NAME AND ADDRESS: BOLT BERANEK AND NEWMAN INC. 21120 Vanowen Street Canoga Park, California 91303	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: 62202F 7231 07 09
11. CONTROLLING OFFICE NAME AND ADDRESS: Aerospace Medical Reseach Laboratory Aerospace Medical Division, Air Force Command, Wright-Patterson, AFB Ohio 45433	12. REPORT DATE: April 1980
	13. NUMBER OF PAGES: 31
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Aircraft Noise
Community Noise Exposure
Environmental Noise Impact

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Noise data for normal aircraft operations are usually calculated by the NOISEMAP computer program at grid points 1000 feet apart. Data from blast noise and supersonic aircraft are calculated at grid points several thousand feet apart. This report describes a computer program that was written to allow the two sets of data to be combined.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

SUMMARY

Noise data for normal aircraft operations are usually calculated by the NOISEMAP computer program at grid points 1000 feet apart. Data from blast noise and supersonic aircraft are calculated at grid points several thousand feet apart. This report describes a computer program that was written to allow the two sets of data to be combined.

PREFACE

This research was performed for the Air Force Aerospace Medical Research Laboratory at Wright-Patterson Air Force Base, Ohio under Project/Task 723107, Technology to Define and Assess Environmental Quality of Noise From Air Force Operations. Technical monitor for this effort was Mr. Jerry D. Speakman of the Biodynamic Environment Branch, Biodynamics and Bioengineering Division.

TABLE OF CONTENTS

INTRODUCTION

Noise data around air bases are normally calculated by NOISEMAP at 1000 foot grid spacings. However, when noise from supersonic aircraft or from blast data is evaluated, it is often desirable to use a much greater grid spacing. The purpose of this task was to provide a means of adding together data calculated at different spacings.

A general technical discussion is given in the next section. Operating instructions are presented in the final section. Appendix A contains a listing of the program. Appendix B contains a sample run including a plot of the resultant grid.

TECHNICAL DISCUSSION

Each set of noise data consists of 10,000 data points. The distance between data points is fixed for any one NOISEMAP computer run, but may be varied between runs. It is desirable to calculate the noise from normal aircraft operations using a relatively small grid spacing. This is required to obtain a sufficient level of detail. Noise from supersonic aircraft or from blasts may cover an extremely large area and a much greater grid spacing may be used. After investigating the two separate components of noise, it may be desirable to combine the data. NOISEMAP does not allow different size grids to be added, so a new procedure was needed.

After investigating several methods of combining the grids, a simple approach using interpolation was found to be the most practical. The output of the program is a new grid of 10,000 points at either of the original grid spacings. There are two basic options available in the program; the small grid may be added to the large grid with the output at the grid spacing of the large grid or the large grid may be added to the small grid with the output at the grid spacing of the small grid.

The program locates the small grid in relationship to the large grid. If the two grids were generated with consistent coordinates, no modifications are required. If required, the origin of either or both of the original dumps may be redefined. The possibility of allowing one grid to be rotated with respect to the other was investigated. The coding required to allow this would be much more complicated and infrequently used. It was therefore decided not to pursue that capability.

The program first reads the two noise dumps produced by NOISEMAP. Each dump contains a header that identifies the following:

- Number of the dump
- Logical unit on which the dump was written
- Type of noise data (i.e. DNL or NEF)
- Date the dump was written
- Title from the 'AIRFLD' card
- Grid spacing
- Field altitude
- X origin
- Y origin
- Magnetic declination.

An array of 100 x 100 noise data points follows the header. Finally, additional data required by NOISEMAP are given.

A new header is then written for the new combined tape. Several items may be changed. The dump number will always be 1. The logical unit number can be defined or will be 14 by default. The date will be the current date. The title is replaced. The grid spacing will be one of the two original grid spacings depending on the option selected. The X and Y origin may be redefined.

If the small grid is added to the large grid, only the area of overlap is affected. If the location of a large grid point coincides with the small grid point, a simple addition is required. If the points do not coincide, a linear interpolation is performed in the X and/or Y direction. Code was prepared to extrapolate the data outside the area of the small grid. In testing the algorithms, it was found that, at times, the noise levels were increasing at the boundary and would, therefore, continue to increase if extrapolated. The purpose of the extrapolation would have been to made sure that the transition appeared smooth. Instead, an area of uncertainty was developed so the code was removed.

Adding the small grid to the large uses the same basic routines. The entire small grid will be updated. A simple linear interpolation is used between large grid points to get the values to be added to the small grid.

OPERATING INSTRUCTIONS

There are five types of control cards that may be required to operate the program. Each type has a keyword and is discussed in detail in the following pages. They follow in alphabetical order but may be inserted into the input deck in any order. Unless otherwise defined, it is assumed that the large grid data are stored on logical unit 12, the small grid data are stored on logical unit 13 and the combined data will be written on logical unit 14.

Of the five cards, only two are required for the program to operate. The first is the TITLE card which provides the airfield title to be written on the header to the dump. The second required card is the OPTION card that determines whether the small grid is added to the large grid or the large grid is added to the small grid.

KEYWORD BGRID

BGRID	12.	100000.	100000.									
1 6	7 14	15 22	23 30	31 38	39 46	47 54	55 62	63 70	71 . . 74	75 . . 78	79 80	

Function: To define the logical unit and X and Y origin for large grid (optional).

Columns

1 - 6	BGRID
7 - 14	Number of the logical unit containing the small grid data (optional). Default value is 12. F 8.0 format.
15 - 22	Redefinition of the X origin of large grid (optional). F 8.0 format. (feet)
23 - 30	Redefinition of the Y origin of large grid (optional). F 8.0 format. (feet)

KEYWORD **CGRID**

CGRID	**14.**										

Function: To define the logical unit for combined grid (optional).

Columns

1 - 5	**CGRID**
7 - 14	Number of the logical unit to be used to write combined grid (optional). Default value is 14. F 8.0 format.

KEYWORD **FGRID**

FGRID	13.	100000.	100000.									

Function: To define the logical unit and X and Y origin for the small grid (optional).

Columns

1 - 5	**FGRID**
7 - 14	Number of the logical unit containing the small grid data (optional). Default value is 13. F 8.0 format.
15 - 22	Redefinition of X origin of small grid (optional). F 8.0 format. (feet)
23 - 30	Redefinition of Y origin of small grid (optional). F 8.0 format. (feet)

KEYWORD **OPTION**

OPTION	**1.**

Function: To identify whether the small grid is added to large grid or large grid is added to small grid.

Columns	
1 - 6	**OPTION**
7 - 14	Large grid added to small grid. Enter 1. Small grid added to large grid. Enter 2.

KEYWORD TITLE

TITLE	EDWARDS AFB COMBINED TAPE

Function: To provide the alpha-numeric data to be used as a title for the airfield.

Columns

1 - 5	TITLE
7 - 76	Alpha-numeric data to be used for the title.

OUTPUT

The printed output from the computer program is very simple. It identifies the tape header for each of the tapes and identifies the option. A sample of the output follows. A dump of noise data is output to the specified logical unit. This dump can then be read by NOISEMAP for developing printed grids or plots.

```
TAPE HEADER FOR LARGE GRID

         DUMP   1 UNIT  13 PROGRAM DNL     DATE  03/29/79
         AIRFIELD EDWARD AFB SUPERSONIC A/C
         GRID SPACING    12000.     FIELD ALTITUDE   2302.
         GRID ORIGIN X -200000.     GRID ORIGIN Y -200000.
         MAGNETIC DECL    14.82

TAPE HEADER FOR SMALL GRID

         DUMP   1 UNIT  13 PROGRAM DNL     DATE  11/17/78
         AIRFIELD EDWARD AFB SUPERSONIC A/C + RANGE
         GRID SPACING     2000.     FIELD ALTITUDE   2302.
         GRID ORIGIN X  292000.     GRID ORIGIN Y  100000.
         MAGNETIC DECL    14.82

TAPE HEADER FOR COMBINED GRID

         DUMP   1 UNIT  14 PROGRAM DNL     DATE  04/09/79
         AIRFIELD EDWARDS AFB COMBINED TAPE
         GRID SPACING     2000.     FIELD ALTITUDE   2302.
         GRID ORIGIN X  492000.     GRID ORIGIN Y  300000.
         MAGNETIC DECL    14.82

****************************************

LARGE GRID ADDED TO SMALL GRID

****************************************
```

SAMPLE OUTPUT

APPENDIX A

COMPUTER PROGRAM LISTINGS

```
      PROGRAM ADDGRD (INPUT, OUTPUT, TAPE5=INPUT, TAPE6=OUTPUT,
     1   TAPE12=514,TAPE13=514,TAPE14=514)
C
C
      REAL KEYWD
      REAL LABEL
      LOGICAL  LBG,LFG
      LOGICAL BGF,FGF
      EQUIVALENCE (IDUMP,HEADER(2))
      EQUIVALENCE (IUNIT,HEADER(3))
      COMMON /GRIDS/ NBF, NBF1, BG(100,100), BS, FG(100,100), FS,
     1  DX, DY, RS, ML, NL, RMM, MM, RNN, NN, T
      DIMENSION ITEXT(7),HEADER(30),LABEL(20),ROW(100)
      DIMENSION CARD(8)
      DATA  TITLE,  FGRID,     BGRID,     OPTION,    CGRID
     1/    5HTITLE,5HFGRID ,  5HBGRID,  6HOPTION, 5HCGRID   /
      DATA LBG,LFG,FGF,BGF /.FALSE.,.FALSE.,.FALSE.,.FALSE./
      DATA T/25.0/
      DATA NI/5/, NO/6/
      DATA NBG,NFG,NCG / 12, 13, 14 /
      DATA NBF /100/
C
C
C      IFLAG IS USED TO INDICATE IF USER WISHES TO GO FROM
C        BIG GRID TO FINE GRID (IFLAG=1) OR FROM FINE GRID TO
C        BIG GRID (IFLAG=2).
C
      REWIND NBG
      REWIND NFG
      CALL DATE(TDATE)
    1 READ(5,5000) CARD
      IF(EOF(5)) 9999,2
    2 CONTINUE
      DECODE(10,5103,CARD) KEYWD
      IF ( KEYWD - TITLE) 3,30
    3 IF ( KEYWD - FGRID ) 4,40
    4 IF ( KEYWD - BGRID ) 5,50
    5 IF ( KEYWD - OPTION) 6,60
    6 IF ( KEYWD - CGRID ) 7,70
    7 WRITE ( 6,6011) KEYWD
      STOP 1
   30 DECODE (70,5101,CARD) (LABEL(I),I=6,15)
      GO TO 1
   40 DECODE (70,5102,CARD) XFG,FX0,FY0
      IF (XFG .EQ. 0.) GO TO 42
      NFG=XFG
   42 CONTINUE
      IF(FX0.NE.0.) GO TO 45
      TEST = SIGN(1.,FX0)
      IF(TEST) 1,45,45
   45 LFG = .TRUE.
      GO TO 1
   50 DECODE (70,5102,CARD) XBG,BX0,BY0
      IF( XBG .EQ. 0.) GO TO 52
      NBG=XBG
   52 CONTINUE
      IF(BX0.NE.0.) GO TO 55
```

```
      TEST = SIGN(1.,BXO)
      IF(TEST) 1,55,55
   55 LBO = .TRUE.
      GO TO 1
   60 DECODE (70,5102,CARD) XFLAG
      IFLAG=XFLAG
      GO TO 1
   70 DECODE (70,5102,CARD) XCG
      NCG=XCG
      GO TO 1
 9999 CONTINUE
      IF(IFLAG.EQ.1) FGF=.TRUE.
      IF(IFLAG.EQ.2) BGF= .TRUE.
  100 READ(NBG) HEADER
      IF(EOF(NBG)) 150,110
  110 IF(HEADER(1).EQ.3HEND) GO TO 150
      PRINT 6000
      PRINT 6010, (HEADER(I),I=1,20)
      BS= HEADER(16)
      IF(LBO) GO TO 112
      BXO= HEADER(18)
      BYO = HEADER(19)
  112 CONTINUE
      HEADER(5) = TDATE
      HEADER(18) = BXO
      HEADER(19) = BYO
      IDUMP=1
      IUNIT=NCG
      DO 113 I=6,15
  113 HEADER(I) = LABEL(I)
      IF (BGF) WRITE(NCG) HEADER
      IF(BGF) PRINT 6002
      IF(BGF) PRINT 6010,(HEADER(I),I=1,20)
      DO 120 J=1,100
      READ(NBG) (BG(I,J),I=1,100)
  120 CONTINUE
  150 READ(NFG) HEADER
      IF(EOF(NFG)) 200,160
  160 IF(HEADER(1) .EQ. 3HEND) GO TO 200
      PRINT 6001
      PRINT 6010 , (HEADER(I),I=1,20)
      FS = HEADER(16)
      IF(LFO) GO TO 162
      FXO = HEADER(18)
      FYO = HEADER(19)
  162 CONTINUE
      HEADER(5) = TDATE
      HEADER(18) = FXO
      HEADER(19) = FYO
      IUNIT=NCG
      IDUMP=1
      DO 163 I=6,15
  163 HEADER(I) = LABEL(I)
      IF(FGF)WRITE(NCG) HEADER
      IF(FGF) PRINT 6002
      IF(FGF) PRINT 6010,(HEADER(I),I=1,20)
      DO 170 J=1,100
```

```
      READ(NFG) (FG(I,J),I=1,100)
  170 CONTINUE
200   CONTINUE
      CALL CALC (BXO, BYO, FXO, FYO)
      IF (IFLAG.EQ.1) GO TO 500
C
C     USE FINE GRID TO GET BIG GRID VALUES
C
      PRINT 6005
      PRINT 6004
      PRINT 6005
      IF ((MM.EQ.ML) .AND. (NN.EQ.NL)) GO TO 300
      CALL INNER
300   CONTINUE
      DO 350 J=1,100
      WRITE(NCG) (BG(I,J),I=1,100)
  350 CONTINUE
      READ(NBG) ITEXT
      WRITE(NCG) ITEXT
      NANN = ITEXT(1)
      IF(NANN) 360,50
  360 DO 370 I=1,NANN
      READ(NBG) ITEXT
      WRITE(NCG) ITEXT
  370 CONTINUE
      STOP 2
C
C
C     USE BIG GRID TO GET FINE GRID VALUES
C
500   CONTINUE
      PRINT 6005
      PRINT 6003
      PRINT 6005
      CALL BGTFG
      DO 400 J=1,100
      WRITE(NCG) (FG(I,J) ,I=1,100)
  400 CONTINUE
      READ(NFG) ITEXT
      WRITE(NCG) ITEXT
      NANN= ITEXT(1)
      IF (NANN) 410,50
  410 DO 420 I=1,NANN
      READ(NFG) ITEXT
      WRITE(NCG) ITEXT
  420 CONTINUE
      STOP 3
C
 5000 FORMAT (8A10)
 5101 FORMAT (6X,10A6,4X)
 5102 FORMAT (6X,8F8.0)
 5103 FORMAT(A6,4X)
 6000 FORMAT (27H1TAPE HEADER FOR LARGE GRID    )
 6001 FORMAT (27H0TAPE HEADER FOR SMALL GRID    )
 6002 FORMAT (30H0TAPE HEADER FOR COMBINED GRID
 6003 FORMAT (31H0LARGE GRID ADDED TO SMALL GRID
 6004 FORMAT (31H0SMALL GRID ADDED TO LARGE GRID
```

```
6005 FORMAT(41H0****************************************  )
6010 FORMAT(1H0, 10X, A4, I4, 6H UNIT I3,  9H PROGRAM A7, 6H DATE  A10/
    2        1H , 10X,  9HAIRFIELD  10A6 /
    3        1H , 10X, 13HGRID SPACING  F9.0,  5X, 15HFIELD ALTITUDE
    4        F7.0  /1H , 10X, 14HGRID ORIGIN X  F8.0, 5X,
    5        14HGRID ORIGIN Y  F8.0 / 1H , 10X, 14HMAGNETIC DECL  F8.2 )
6011 FORMAT (19H INVALID KEYWORD    ,A10)
     END
```

```
      SUBROUTINE CALC (BX0, BY0, FX0, FY0)
      COMMON /GRIDS/ NBF, NBF1, BG(100,100), BS, FG(100,100), FS,
     1   DX, DY, RS, ML, NL, RMM, MM, RNN, NN, T
C
C
C     NBF = NUMBER OF ROWS (COLUMNS) IN THE GRIDS
C     NBF1 = NBF - 1
C     BG = ARRAY OF BIG GRID POINTS
C     BS = SPACING IN BIG GRID
C     BX0,BY0 - X,Y COORDINATES FOR ORIGIN OF BIG GRID
C     FG - ARRAY OF FINE GRID VALUES
C     FS = SPACING FOR FINE GRID
C     FX0,FY0 - X,Y COORDINATES FOR ORIGIN OF FINE GRID
C     RS = BS / FS
C     DX -    FX0 = BX0 + (DX-1)*BS
C     DY -   FY0 = BY0 + (DY-1)*BS
C     ML,NL - INDICES OF LOWEST X,Y COORDINATES OF THOSE BIG GRID
C       POINTS CONTAINED IN FINE GRID
C     MM,NN - INDICES OF HIGHEST X,Y COORDINATES OF THOSE BIG GRID
C       POINTS CONTAINED IN FINE GRID
C
C
      NBF1 = NBF - 1
      DX = (FX0-BX0) / BS + 1.
      DY = (FY0-BY0) / BS + 1.
      RS = BS/FS
      ML = IFIX(DX)
      NL = IFIX(DY)
      R99 = NBF1 / RS
      RMM = DX + R99
      RNN = DY + R99
      MM = IFIX(RMM)
      NN = IFIX(RNN)
      IF (DX-ML .NE. 0) ML = ML + 1
      IF (DY-NL .NE. 0.) NL = NL + 1
      RETURN
      END
```

```
      SUBROUTINE INNER
C
C     INNER LOOPS ON ALL BIG GRID POINTS CONTAINED IN THE FINE GRID
C       INTERPOLATING FINE GRID POINTS TO GET VALUES FOR EACH.
C
      COMMON /GRIDS/ NBF, NBF1, BG(100,100), BS, FG(100,100), FS,
     1  DX, DY, RS, ML, NL, RMM, MM, RNN, NN, T
C
      YJ = (NL-DY-1.) * RS + 1.
      DO 500 J=NL,NN
        YJ = YJ + RS
        JY = IFIX(YJ)
        XI = (ML-DX-1.) * RS + 1.
        DO 400 I=ML,MM
          XI = XI + RS
          IX = IFIX(XI)
          CALL INTERF(IX,JY,XI,YJ,G)
          BG(I,J) = BG(I,J) + G
400     CONTINUE
500   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE INTERF (I, J, RI, RJ, G)
C
C     INTERF INTERPOLATES FINE GRID POINTS TO GET A SINGLE VALUE
C       FOR A BIG GRID POINT.
C       I,J ARE THE INDICES  FOR THE FINE GRID POINT CLOSEST,
C       BUT TO THE LEFT AND BELOW, THE DESIRED BIG GRID POINT.
C       RI,RJ ARE THE ACTUAL FLOATING POINT COORDINATES THE BIG
C       GRID POINT WOULD HAVE WERE IT IN THE FINE GRID.
C
      COMMON /GRIDS/ NBF, NBF1, BG(100,100), BS, FG(100,100), FS,
     1  DX, DY, RS, ML, NL, RMM, MM, RNN, NN, T
C
C     INTERPOLATE IN X DIRECTION
C
      IF (RI .EQ. I) GO TO 200
      GX = (FG(I+1,J) - FG(I,J)) * (RI-I) + FG(I,J)
      IF (RJ .NE. J) GO TO 400
      G = GX
      RETURN
C
C     NO INTERPOLATION NEEDED
C
200   CONTINUE
      IF (RJ .NE. J) GO TO 300
      G = FG(I,J)
      RETURN
C
C     INTERPOLATE IN Y DIRECTION
C
300   CONTINUE
      G = (FG(I,J+1) - FG(I,J)) * (RJ-J) + FG(I,J)
      RETURN
C
C     INTERPOLATE IN BOTH DIRECTIONS
C
400   CONTINUE
      GY = (FG(I+1,J+1) - FG(I,J+1)) * (RI-I) + FG(I,J+1)
      G = (GY - GX) * (RJ-J) + GX
      RETURN
      END
```

```
      SUBROUTINE INTERB (V1,V2,D,R,DE,G,NG)
C
C     INTERB INTERPOLATES BIG GRID  POINTS TO GET VALUES FOR
C       FINE GRID POINTS
C       V1, V2   -  GRID VALUES TO BE INTERPOLATED
C       D      -  DISTANCE (IN BIG GRID SPACING) TO FIRST FINE GRID POINT
C                 FROM V1
C       R      -  FS / BS
C       DE     -  OUTPUT VALUE DISTANCE FROM V2 TO NEXT FINE GRID POINT
C       G      -  ARRAY OF INTERPOLATED VALUES
C       NG     -  NUMBER OF VALUES IN G
C
      DIMENSION G(1)
C
C
      C = V2 - V1
      G(1) = V1 + C*D
      C = C *R
      IF (NG .NE. 0) GO TO 300
      NG = 1
      DE = D
      IF (DE .GE. 1) RETURN
C
100   CONTINUE
        DE = DE + R
        IF (DE .GT. 1.) RETURN
        NG = NG + 1
        G(NG) = G(NG-1) + C
      GO TO 100
C
C
300   CONTINUE
      IF (NG .EQ. 1) RETURN
      DO 400 I=2,NG
        G(I) = G(I-1) + C
400   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE BGTFG
C
C
C     BGTFG  USES BIG GRID POINTS TO INTERPOLATE VALUES FOR ALL FINE
C       GRID POINTS.
C
      COMMON /GRIDS/ NBF, NBF1, BG(100,100), BS, FG(100,100), FS,
     1  DX, DY, RS, ML, NL, RMM, MM, RNN, NN, T
      COMMON /SCRACH/ G1(100), G2(100), G3(100)
C
C     INITIAL COMPUTATIONS
C
      MML = ML - 1
      IF (ML .EQ. DX) MML = ML
      NNL = NL -1
      IF (NL .EQ. DY) NNL = NL
      MMM = MM + 1
      IF (MM .EQ. RMM) MMM = MM
      NNN = NN
      IF (NN .EQ. RNN) NNN = NN - 1
      MP1 = MML + 1
      XDE = 0.
      YDE = 0.
      XDO = DX - MML
      YD = DY - NNL
      RRS = 1. / RS
      LJO = 0
C
C     LOOP ON BIG GRID POINTS THAT SURROUND FINE GRID
C
C
      DO 800 J=NNL,NNN
        NG1 = 0
        CALL INTERB(BG(MML,J),BG(MML,J+1),YD,RRS,YDE,G1,NG1)
        NG2 = NBF - LJO
        NG1 = MIN0(NG1,NG2)
        NG2 = NG1
        XD = XDO
        LIO = 0
        DO 600 I=MP1,MMM
          CALL INTERB (BG(I,J),BG(I,J+1),YD,RRS,YDE,G2,NG2)
          N = NBF - LIO
          NG3 = 0
          LJ = LJO
          DO 400 K=1,NG2
            CALL INTERB (G1(K),G2(K),XD,RRS,XDE,G3,NG3)
            LJ = LJ + 1
            NG3 = MIN0(N,NG3)
            LI = LIO
            DO 300 L=1,NG3
              LI = LI + 1
              FG(LI,LJ) = FG(LI,LJ) + G3(L)
300         CONTINUE
400       CONTINUE
          DO 500 K=1,NG2
            G1(K) = G2(K)
500       CONTINUE
```

```
                  XD = XDE - IFIX(XDE)
                  LIO = LI
        600     CONTINUE
                LJO = LJ
                YD = YDE - IFIX(YDE)
        800   CONTINUE
              RETURN
              END
```

APPENDIX B

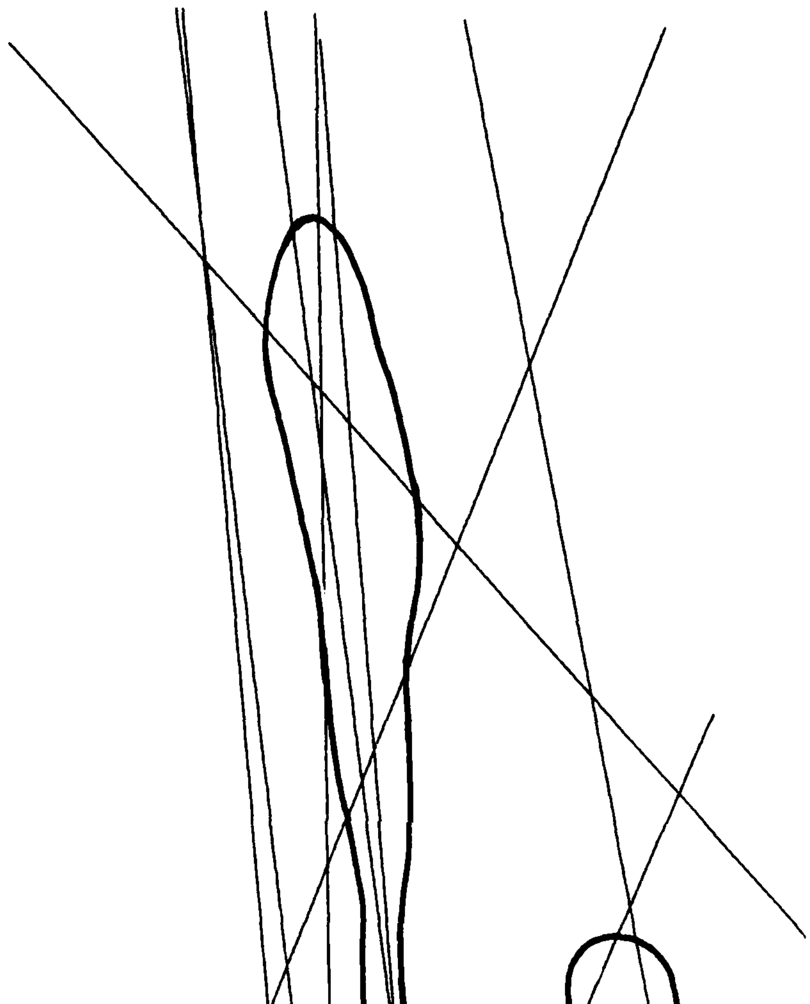
SAMPLE COMPUTER RUN

```
TITLE EDWARDS AFB COMBINED TAPE
OPTION2.
BGRID 12.     0.      0.
FGRID 13.     492000. 300000.
```

EXAMPLE INPUT DATA

```
TAPE HEADER FOR LARGE GRID

          DUMP    1 UNIT  13 PROGRAM DNL      DATE  03/29/79
          AIRFIELD EDWARD AFB SUPERSONIC A/C
          GRID SPACING     12000.      FIELD ALTITUDE   2302.
          GRID ORIGIN X -200000.      GRID ORIGIN Y -200000.
          MAGNETIC DECL     14.82

TAPE HEADER FOR COMBINED GRID

          DUMP    1 UNIT  14 PROGRAM DNL      DATE  04/18/80
          AIRFIELD EDWARDS AFB COMBINED TAPE
          GRID SPACING     12000.      FIELD ALTITUDE   2302.
          GRID ORIGIN X          0.      GRID ORIGIN Y          0.
          MAGNETIC DECL     14.82

TAPE HEADER FOR SMALL GRID

          DUMP    1 UNIT  13 PROGRAM DNL      DATE  11/17/78
          AIRFIELD EDWARD AFB SUPERSONIC A/C + RANGE
          GRID SPACING      2000.      FIELD ALTITUDE   2302.
          GRID ORIGIN X  292000.      GRID ORIGIN Y  100000.
          MAGNETIC DECL     14.82

****************************************

SMALL GRID ADDED TO LARGE GRID

****************************************
```

EXAMPLE OUTPUT DATA



SAMPLE COMPUTER GRAPHIC OUTPUT

*U.S.Government Printing Office: 1980 — 657-084/778